

Claims

WHAT IS CLAIMED IS:

- Sub A1  
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
1. A computing system having a mass storage device and a system timer for obtaining benchmark timing for a portion of an application program execution, the computing system comprising:
    - 4 a mass storage system;
    - 5 an init module for determining if the timestamp data is to be collected during the operation of the application program;
    - 7 a performance marker module for obtaining and storing the timestamp data for later retrieval;
    - 9 an uninit module for formatting and storing the obtained timestamp data into a data file within the mass storage device that permits retrieval after the termination of the application program; and
    - 12 a performance benchmark data post processing module for determining the benchmark timing from two or more timestamp data entries;
    - 14 wherein the performance marker module is executed at predefined points within a plurality of processing modules within the application program.

- Sub B2  
1 2 3 4 5 6
1. 2. The system according to claim 1, wherein:
    - 2 the init module is executed before any timestamp data is collected;
    - 3 the performance marker module is executed each time benchmark timestamp data and overhead timestamp data is to be collected;
    - 5 the uninit module is executed after all timestamp data desired has been collected to store the timestamp data within records of a Raw Data Table; and

7           the performance benchmark data post processing module determines the  
8           benchmark timing from the records stored within the Raw Data Table.

1           3.       The computing system according to claim 2, wherein the init module determines if  
2           timestamp data is to be collected.

1           4.       The computing system according to claim 3, wherein init module makes the  
2           determination that timestamp data is to be collected by checking for the existence of an  
3           identification key within a system registry;

4           the identification key uniquely identifies the processing modules to be used to collect,  
5           format, and store the run-time internal state data to be collected.

1           5.       The computing system according to claim 4, wherein the performance marker  
2           module collects timestamp data only if the init module has determined that the timestamp data is  
3           to be collected.

1           6.       The computing system according to claim 5, wherein the performance marker  
2           module generates a data record within the Raw Data Table each time the performance marker  
3           module is executed.

1           7.       The computing system according to claim 6, wherein the benchmark data record  
2           further containing an overhead timestamp data value each time the performance marker module is  
3           executed.

1           8.       The computing system according to claim 7, wherein the performance marker  
2           module stores the benchmark data records within a data memory block within the processing  
3           modules identified by an identification key within a system registry.

b2  
cmx  
PCT/US2016/036699

1           9. The computing system according to claim 8, wherein the ununit module retrieves  
2 the data records from the data memory block for transfer to the Raw Data Table on the mass  
3 storage device.

1           10. The computing system according to claim 9, wherein the performance benchmark  
2 data post processing module determines the benchmark timing from difference between two  
3 benchmark timestamp data entries stored within the Raw Data Table to generate a second data  
4 Record within a Processed Data Table.

1           11. The computing system according to claim 10, wherein the data record within the  
2 Raw Data Table comprises a ResultID field, an AppID field, a MarkerID field, a Marker Cycles  
3 Field, and an Overhead Cycles field.

1           12. The computing system according to claim 11, wherein the second data record of  
2 the Processed Data Field comprises a ResultID field, a Reboot Iteration field, a Launch Iteration  
3 field, a Marker Iteration field, a Marker Pair ID field, and a Seconds Field.

1        13. The computing system according to claim 12, wherein the ResultID field of the  
2 Raw Data Table corresponds to the ResultID field in the Processed Data Table.

3           14. The computing system according to claim 12, wherein the Marker Pair ID field  
4 corresponds to a second Marker Pair ID field in a Marker Pair Table.

5        15. The computing system according to claim 14, wherein the Marker Pair Table  
6 comprises the second Marker Pair ID field, a start App ID field, a start Marker ID field, an End  
7 App ID field, an End Marker ID field, and a MarkerPair Name field.

1           16. A method for obtaining benchmark timing for a portion of an application program  
2 execution, the method comprising:

3           inserting one or more code markers into the application program at predefined locations  
4       within the application program corresponding to the point at which benchmark timing data is  
5       desired;

6           determining if benchmark timing data is to be collected at each code marker by checking  
7       for the existence of processing modules identified by an identification key within a system  
8       registry;

9           if benchmark timing data is to be collected at each code marker:

10           generating a benchmark data record containing the collected benchmark timing  
11       data each time the code markers are reached;

12           storing the benchmark data records within a data memory block within the  
13       processing modules identified by the identification key within the system registry;

14           retrieving the benchmark data records from the data memory block for transfer to  
15       first data record in a Raw Data Table device once all of the run-time internal state data has  
16       been collected; and

17           processing the first data records stored within the Raw Data Table to generate  
18       second data records in a Processed Data Table that estimate the benchmark timing defined  
19       between two benchmark data records.

1           17.     The method according to claim 16, wherein the benchmark timing generated and  
2       stored within the processed data table is determined from difference between two benchmark  
3       timestamp data entries stored within the raw data table.

1           18.     The method according to claim 17, wherein  
2       the benchmark timing is determined by subtracting an estimate for the total overhead  
3       processing from the difference between two benchmark timestamp data entries stored within the  
4       raw data table;

5       the estimate for the total overhead processing is determined by totaling the difference  
6       between an overhead timestamp value and a benchmark timestamp value for all code markers  
7       between the two benchmark timestamp entries used to determine the benchmark timing;  
8              the benchmark timestamp value is obtained from a system timer immediately after a code  
9       marker is reached; and  
10          the overhead timestamp value is obtained from the system timer immediately before the  
11       processing returns to the application program from performance marker processing.

*Sub  
By  
end*

1       19.     The method according to claim 18, wherein the first data record within the Raw  
2       Data Table comprises a ResultID field, an AppID field, a MarkerID field, a Marker Cycles Field,  
3       and an Overhead Cycles field.

1       20.     The method system according to claim 19, wherein the second data record of the  
2       Processed Data Field comprises a ResultID field, a Reboot Iteration field, a Launch Iteration  
3       field, a Marker Iteration field, a Marker Pair ID field, and a Seconds Field.

*Sub  
A4*

1       21.     The method according to claim 22, wherein the ResultID field of the Raw Data  
2       Table corresponds to the ResultID field in the Processed Data Table.

*Sub  
B4*

3       22.     The method according to claim 21, wherein the Marker Pair ID field corresponds  
4       to a second Marker Pair ID field in a Marker Pair Table.

5       23.     The method according to claim 22, wherein the Marker Pair Table comprises the  
6       second Marker Pair ID field, a start App ID field, a start Marker ID field, an End App ID field, an  
7       End Marker ID field, and a MarkerPair Name field.

Sub A1

1        24. A computer data product readable by a computing system and encoding a  
2        computer program of instructions for executing a computer process for obtaining run-time  
3        internal state data within an application program, said computer process comprising:  
4                inserting one or more code markers into the application program at predefined locations  
5        within the application program corresponding to the point at which benchmark timing data is  
6        desired;  
7                determining if benchmark timing data is to be collected at each code marker by checking  
8        for a processing modules identified by an identification key within a system registry;  
9                if benchmark timing data is to be collected at each code marker:  
10                        generating a benchmark data record containing the collected benchmark timing  
11        data each time the code markers are reached;  
12                        storing the benchmark data records within a data memory block within the  
13        processing modules identified by the identification key within the system registry;  
14                        retrieving the benchmark data records from the data memory block for transfer to  
15        first data record in a Raw Data Table device once all of the run-time internal state data has  
16        been collected; and  
17                        processing the first data records stored within the Raw Data Table to generate second data  
18        records in a Processed Data Table that estimate the benchmark timing defined between two  
19        benchmark data records;  
20                        wherein the benchmark timing generated and stored within the processed data table is  
21        determined from difference between two data entries stored within the raw data table.

Sub B8

---

1        25. The computer data product according to claim 24, wherein the benchmark timing  
2        is determined by subtracting an estimate for the total overhead processing from the difference  
3        between two benchmark timestamp data entries stored within the raw data table;

4           the estimate for the total overhead processing is determined by totaling the difference  
5        between an overhead timestamp value and a benchmark timestamp value for all code markers  
6        between the two benchmark timestamp entries used to determine the benchmark timing;  
7           the benchmark timestamp value is obtained from a system timer immediately after a code  
8        marker is reached; and  
9           the overhead timestamp value is obtained from the system timer immediately before the  
10      processing returns to the application program from performance marker processing.

*Sub*  
*B8*  
*Cin*

1       26.     The computer data product according to claim 25, wherein the first data record  
2       within the Raw Data Table comprises a ResultID field, an AppID field, a MarkerID field, a  
3       Marker Cycles Field, and an Overhead Cycles field.

1       27.     The computer data product system according to claim 26, wherein the second data  
2       record of the Processed Data Field comprises a ResultID field, a Reboot Iteration field, a Launch  
3       Iteration field, a Marker Iteration field, a Marker Pair ID field, and a Seconds Field.

1       28.     The computer data product according to claim 27, wherein the ResultID field of  
2       the Raw Data Table corresponds to the ResultID field in the Processed Data Table.

1       29.     The computer data product according to claim 28, wherein the Marker Pair ID  
2       field corresponds to a second Marker Pair ID field in a Marker Pair Table.

1       30.     The computer data product according to claim 29, wherein the Marker Pair Table  
2       comprises the second Marker Pair ID field, a start App ID field, a start Marker ID field, an End  
3       App ID field, an End Marker ID field, and a MarkerPair Name field.

1        31. The comp[...] data product according to claim 23, wherein the computer data  
2        product comprises a computer readable storage medium readable by a computer upon which  
3        encoded instructions used to implement the computer process are stored.

*Sub  
B8  
ord*

00000000000000000000000000000000